Article

Machine Learning-Based Network Sub-Slicing Framework in a Sustainable 5G Environment

Sushil Kumar Singh [1] **, Mikail Mohammed Salim** [1] **, Jeonghun Cha** [1] **, Yi Pan** [2] **and Jong Hyuk Park** [1,*]

[1]   Department of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul 01811, Korea; sushil.sngh001007@seoultech.ac.kr (S.K.S.); mikail@seoultech.ac.kr (M.M.S.); ckwjdgns@seoultech.ac.kr (J.C.)
[2]   Department of Computer Science, Georgia State University, Atlanta, GA 30302-5060, USA; yipan@gsu.edu
*   Correspondence: jhpark1@seoultech.ac.kr; Tel.: +82-2-970-6702

check for updates

Abstract: Nowadays, 5G network infrastructures are being developed for various industrial IoT (Internet of Things) applications worldwide, emerging with the IoT. As such, it is possible to deploy power-optimized technology in a way that promotes the long-term sustainability of networks. Network slicing is a fundamental technology that is implemented to handle load balancing issues within a multi-tenant network system. Separate network slices are formed to process applications having different requirements, such as low latency, high reliability, and high spectral efficiency. Modern IoT applications have dynamic needs, and various systems prioritize assorted types of network resources accordingly. In this paper, we present a new framework for the optimum performance of device applications with optimized network slice resources. Specifically, we propose a Machine Learning-based Network Sub-slicing Framework in a Sustainable 5G Environment in order to optimize network load balancing problems, where each logical slice is divided into a virtualized sub-slice of resources. Each sub-slice provides the application system with different prioritized resources as necessary. One sub-slice focuses on spectral efficiency, whereas the other focuses on providing low latency with reduced power consumption. We identify different connected device application requirements through feature selection using the Support Vector Machine (SVM) algorithm. The K-means algorithm is used to create clusters of sub-slices for the similar grouping of types of application services such as application-based, platform-based, and infrastructure-based services. Latency, load balancing, heterogeneity, and power efficiency are the four primary key considerations for the proposed framework. We evaluate and present a comparative analysis of the proposed framework, which outperforms existing studies based on experimental evaluation.

Keywords: machine learning; network sub-slicing; sustainable 5G environment; IoT

1. Introduction

Communication has become an essential part of our daily lives in a sustainable society. 5G has emerged as a key technology in enabling a wide range of sustainable development goals, from good health to energy efficiency, and access to a sustainable environment. Today, many types of communication are available including Human to Machine (H2M), Machine to Machine (M2M), and Peer to Peer (P2P) computing. 5G networks enable humans to interconnect with machines, objects, and IoT (Internet of Things) devices. The potent combination of 5G, artificial intelligence, smart platforms, and IoT will change the world, offering intelligent, sustainable connectivity to diverse services as applications, platforms, and infrastructures. The objective of the 5G network infrastructure is to provide opportunities to generate new services and new business strategies for

entering communication networks [1]. The IoT has many applications in diverse fields, such as smart healthcare, smart transportation, smart agriculture, and industrial automation, making use of the connectivity of devices. Therefore, 5G network connectivity offers the ability to enhance these applications and enables new ones—such as automotive transportation and agriculture—with machine learning. Nonetheless, these applications generate massive amounts of data in the cloud network. According to a report released by Gartner [2], an end-to-end sustainable 5G infrastructure will be converted by 2025 to 2030 and deployed in most IoT and Artificial Intelligence (AI) applications.

IoT devices are a combination of numerous sensors in vehicles, medical devices, RFID tags, and other embedded devices that communicate, transmit, and exchange data to serve user applications. Diverse domains in smart cities such as smart healthcare, transportation, buildings, and industrial automation benefit from data derived from IoT devices to provide optimized services to users. The quick deployment of embedded sensor devices accelerated with the increase in the adoption of 5G technology across different nations. However, with large amounts of data generated, the optimal provision of bandwidth for exchange and transmission of data between devices is challenging even for the latest network technologies.

Ever evolving, growing business models and smart cities pose a challenge to the existing monolithic network architecture. A traditional network architecture such as LTE has high latency, low data rates, and inadequate coverage. However, a slicing-enabled sustainable network architecture in 5G allows the system to meet the diverse and vital demands of different applications. Network slicing is a logical, virtualization-based division of the physical resources configured to meet various vertical industry requirements. Traditional LTE-based systems entail the construction of individual networks for applications with special needs, such as low latency, high reliability, or widespread device implementation [3]. It is, however, not feasible to slice a network based on standard and common features for every application of the IoT in a sustainable 5G environment. For example, an application that requires high reliability with very low latency may not be capable of providing spectral efficiency [4].

Network slicing is an essential method for utilizing network services in the sustainable 5G environment. As the embodiment concept in running various logical networks has virtually independent IoT operations on a standard physical environment [5,6], it has the advantage of virtualizing infrastructures wherein multiple applications are used on the physical cloud networks [7]. Virtualized resources allow the flexible, efficient utilization of limited resources and provide maximum services, essentially mobile and IoT, among others. Still, it has certain limitations related to centralization, load balancing, high latency, and the handling of large amounts of data [8,9].

The existing 5G network architecture as shown in Figure 1, allows a physical node to provide services to separate tenants based on their requirements. These tenants provide individual services based on their configuration. Network slicing is accomplished using software solutions such as Software-defined networking (SDN) and Network Functions Virtualization (NFV) [10]. The SDN and NFV techniques have such properties as programmability and modularity. New multiple logical networks are created by using resources. The physical layer is a standard layer that supports various other logical sliced networks by using the radio access network, which includes core networks such as mobile exchange and data networks [11].

Although network slices in a sustainable 5G network aim to reduce the load on a single physical cloud-based resource, the existing architecture is unable to meet modern application demands [12]. Smart city-based applications, including Smart Manufacturing, Smart Vehicles, and Smart Grid need a wide range of devices with different network requirements and configurations. For example, in hospitals, there are multiple low compute-intensive devices such as weight scale and blood pressure cuffs for cancer treatment. These devices require low-latency, high-reliability network configurations. On the other hand, database and device security systems are high computing-intensive systems requiring high bandwidth to monitor multiple devices spread in different locations [13,14]. A single slice of network based on a standard configuration for the entire hospital would not be able to deliver the performance needed to manage the system and devices in an optimum manner.
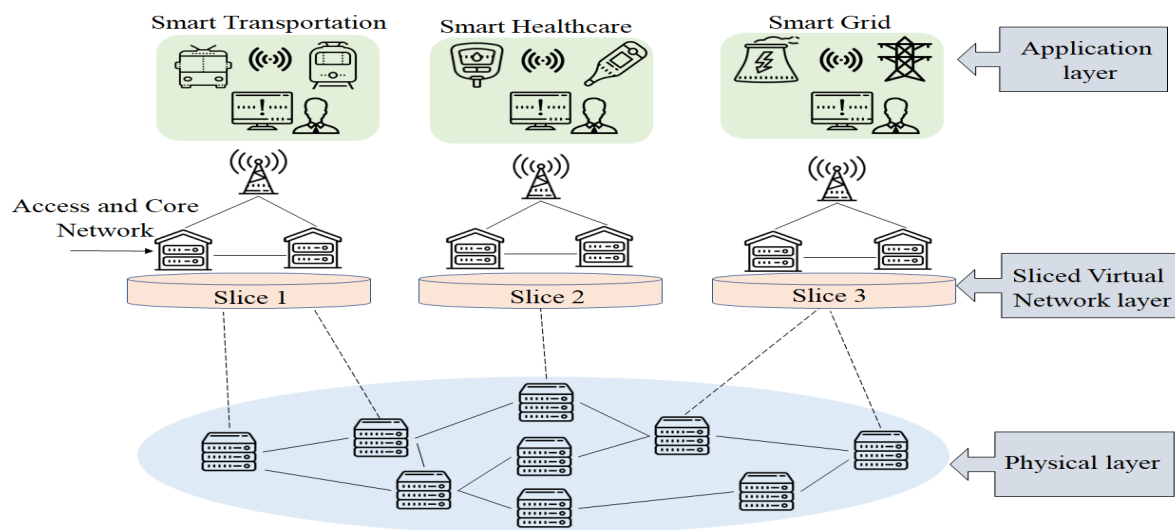
Figure 1. 5G Network slicing architecture.

Existing studies suggest measures for maintaining the allocation of network slices to different applications efficiently. The aim of the present paper is to address load balancing issues in dynamic applications running in a sustainable 5G network environment. Each slice has a finite amount of resources, so services need to be allotted efficiently to ensure the optimum performance of the network. We introduce the concept of sub-slicing in a 5G network environment based on Machine Learning, where resources are allocated based on the varying demand for services such as application-based, platform-based, and infrastructure-based services. Machine learning-based algorithms are used to learn the features of the application as well as in which areas in the network its resources must be assigned, such as lower latency or higher spectral efficiency [15,16]. We use the Support Vector Machine (SVM) algorithm to learn the features of different application-based devices and employ the k-means algorithm to cluster the set of devices efficiently based on their common needs or services and to assign the group based on similar types of application services, such as application-based, platform-based, and infrastructure-based services. Every sub-slice provides suitable frequencies according to the requirement of the applications, i.e., whether they are hardware-based or software-based. The proposed framework improves bandwidth consumption in a 5G network slice. The reduced usage of data rate by dividing an application into sub-modules optimizes the throughput of each cluster. As each sub-module has different network resource requirements, proper allocation of the data rate ensures efficient utilization of a limited spectrum in the proposed framework.

To summarize, the main contribution of this paper is as follows:

- We propose the Machine Learning-based Network Sub-slicing Framework in a sustainable 5G environment where the SVM algorithm is used for feature selection based on IoT devices. The K-Means algorithm is utilized to assign the group based on similar types of application services, such as application-based, platform-based, and infrastructure-based services.
- Network Sub-slicing and related studies are described in detail to mitigate the problems of resource allocation in a sustainable 5G environment of the network. The key requirements and security threats for a Network Sub-slicing framework are identified to achieve the optimum allocation of resources or services.
- We present an overview and the detailed methodology of the proposed Machine Learning-based Network Sub-slicing Framework for enabling the efficient, optimal usage of network resources.
- Finally, we conduct an evaluation and an experimental analysis to demonstrate the effectiveness of the proposed network sub-slicing framework compared to existing research.

The rest of this paper is organized as follows: Section 2 discusses related research based on the efficient allocation of resources using network slicing as the seminal contribution and key consideration; Section 3 describes the proposed Machine Learning-based Network Sub-slicing Framework in a sustainable 5G environment and the methodological flow for the sub-slicing of the network slice using machine learning algorithms such as SVM and k-means; in Section 4, we design a prototype and present an analysis of the proposed framework that outperforms existing studies, based on the experimental evaluation; Section 5 presents the conclusion.

2. Related Work

In this section, we first discuss the contribution of existing studies with regard to network slicing in the 5G environment. Second, we present the challenges posed by latency, load balancing, centralization, heterogeneity, and energy efficiency, which are addressed in the proposed Machine Learning-based Network Sub-slicing Framework in a 5G environment. Several existing studies addressed network slicing in 5G; nonetheless, there are still open challenges for sustainable 5G-enabled IoT applications. In this section, we further discuss the security threats for network slicing in the 5G environment.

2.1. Seminal Contributions

Many researchers have examined network slicing concepts and provided customers with configurable services that can achieve more significant economies of scale and efficiency for multiple applications such as healthcare, transportation, farming, and other IoT applications. Nour et al. [17] proposed a broker design mechanism based on permissionless blockchain technology and provided secure and anonymous transactions in the 5G network. The network slice provider allows the brokering mechanism to lease its resources from various providers in a safe manner. However, the proposed tool for network slicing has limitations in selecting the resource orchestration interfaces. Sciancalepore et al. [18] designed blocks for network slicing, which have four features: (1) traffic and user mobility analysis; (2) learning and forecasting schemes per slice; (3) optimal admission control; (4) a reinforcement process to drive the system. The RL-NSB infrastructure performed a service level agreement with various tenants as traffic usage and user distribution for admission control and enhanced the overall process by reinforcement learning. The network slice scheduling keeps track of the service level agreement for different slices, and then transfers this information back to the forecasting module and adjusts the system behavior.

Aviles et al. [19] proposed a practical open-source solution for end-to-end network slicing (POSENS) in mobile networks. They developed an end-to-end network slicing protocol stack that provided Radio Access Network (RAN) solutions with slice awareness. POSENS includes multi-slice user equipment ability for multiple slice management and orchestration capabilities for shared RAN solutions, as well as supporting independent, customizable network slices. Hao et al. [20] introduced the network slice-based 5G wearable network, which is a combination of the 5G ultra-dense cellular network, edge caching, and edge computing. The primary aim of the proposed data-driven resource management framework is to manage and realize network slicing resources efficiently in 5G networks. A hierarchical cognitive engine architecture was used to conduct data-driven resource management, with a cognitive service engine utilized to analyze the service features. The global cognitive engine was used to achieve service-aware resource allocation. Dighriri et al. [21] proposed a novel resource allocation scheme for different slices to satisfy diverse users' requirements. The network slice communicates the desirable quality of service (QoS) level to the admission control entity, and the admitted slices are allocated physical radio resources by the virtual network based on intra- and inter-slice priority.

Kasgari et al. [22] developed a novel framework using stochastic optimization to enable 5G network slicing with effective isolation. To address Reliable Low Latency (RLL) and self-managed slice problems, the authors used a stochastic optimization novel framework based on the Lyapunov drift-plus-penalty method. The structure provides reliable, low latency-based, end-to-end communication for RLL slices. Bega et al. [23] presented DeepCog, a novel data analytics tool for the cognitive management of resources

in 5G networks. This tool aims to maintain a balance between the over-provisioning of resources and service request violations. The deep learning-based architecture was designed for capacity forecasting based on the presented experimental evaluation using real-world data. The DeepCog method is trained through customizable loss function to predict mobile traffic in 5G networks. Abbas et al. [24] proposed the mobile central office R-architected as a data center (M-Cord). They introduced a new slicing mechanism for the transport network and it used in between the access and core network. Kafle et al. [25] introduced the envisioned 5G network slicing and elaborated on the need to automate network functions for the design, construction, deployment, operation, control, and management of network slices. They used Machine learning SVM and K-Means algorithms. Le et al. [26] studied the integration of various machine learning algorithms, big data, SDN, and NFV, proposed a comprehensive architecture for network slicing in 5G, and implemented it in a traffic classification with SVM.

2.2. Key Considerations for 5G Network Sub-Slicing

The four key considerations in establishing an effective method of network sub-slicing in sustainable 5G environments are as follows: latency, load balancing, heterogeneity, and power efficiency. The importance of each of the identified key considerations is described below.

- Latency: This is one of the Network Slicing concepts used in 5G for utilizing IoT applications such as smart parking, smart roads, and smart manufacturing. One network slice is sufficient to handle one IoT application. However, as one application is not a small part of one network slice, service latency will be high in one IoT application for 5G Networks. This then is the main consideration in the sustainable 5G network environment for IoT applications. To mitigate this problem, we used the concept of network sub-slicing in 5G. We show the evaluation of the latency graph with the number of services and running time parameters and compare them to the existing framework of research.

- Load balancing: *Load balancing* entails efficiently distributing incoming network traffic across a group of backend servers, also known as a server pool. By spreading the work evenly, load balancing improves the availability of applications. It is also the main consideration here because one network slice cannot handle the load of one IoT application, in a sustainable 5G environment. In the virtualized environment of 5G networks, the control and management of various applications such as smart healthcare and smart vehicles are handled by dynamic network slices. Therefore, load balancing is an issue in network slices. Nonetheless, these applications have multiple network services (application-based, infrastructure-based, and platform-based) that cannot be managed by one network slice, including load balancing issues. Thus, we divided one network slice into various network sub-slices. These network sub-slice mechanisms provide a particular service to IoT applications through the decentralized setting of point of presence (PoP). Each PoP has knowledge of the aggregate workload generated by the slice and users accessing it.

- Heterogeneity: 5G wireless network applications generate high demand for data rates to utilize all IoT applications. Thus, every network is divided into various network slices according to the requirements for IoT applications. It is called a "heterogeneous network". Still, all network slices are unable to handle all the needs of a particular IoT application. Thus, the IoT application features are divided into subparts according to services such as application-based, infrastructure-based, and platform-based services. For example, if we have one IoT application for smart healthcare, it has various features such as IP address, types of networks, types of protocols, and the number of packets for sending data, among others. These features are classified according to application-based, infrastructure-based, and platform-based services.

- Power efficiency: Power efficiency refers to the ratio of power divided into the input power according to the requirement. Advanced IoT applications have dynamic bandwidth requirements. Thus, the proposed framework allocates the spectrum based on the varying demands of the process of application services such as application-based (ABS), platform-based (PBS),

and infrastructure-based services (IBS) (Amazon web services, Google App Services, DropBox, and others). These services may be software-based or hardware-based. A sustainable 5G environment is then provided for advanced applications with the help of network sub-slicing. Power efficiency is a major requirement for both network service providers and end users. The higher data rate consumption in 5G networks results in faster battery consumption for users and higher operating costs for service providers; hence the need for energy-efficient solutions. The input power to measure is in Mbps/W, where W stands for Watt, and describes the relationship between the energy consumed and the bandwidth of data transmitted.

Based on the analysis of the studies mentioned above, there are currently no methods of improving network slicing for a sustainable 5G environment. Network slicing needs to be further categorized in various subslices to mitigate the above-mentioned challenges and provide optimum performance of IoT devices in 5G. As such, we present the concepts of network sub-slicing in the 5G environment with the help of machine learning algorithms. In this paper, we propose a machine learning-based network sub-slicing framework in a sustainable 5G environment and mitigate all of the above-mentioned issues.

3. Proposed Network Sub-Slicing Framework in a Sustainable 5G Environment

The existing studies examine various considerations such as low latency, management of IoT services, centralization, and computational cost for network slicing in 5G-enabled IoT applications. In this section, we present a Machine Learning-based Network Sub-slicing Framework in a sustainable 5G environment and mitigate the limitations of existing studies. The SVM algorithm is utilized for feature selection based on IoT devices, and the K-Means algorithm is used to assign the group based on similar types of application services such as application-based, platform-based, and infrastructure-based services. These services are then managed according to a server, network, operating system, data control, storage management, testing devices, cloud computing, and other factors.

3.1. Design Overview of the Proposed Network Sub-Slicing Framework

The 5G environment is a new paradigm in the provision of faster, higher-capacity transmission of massive amounts of data related to IoT applications, such as smart healthcare, smart transportation, and smart farming apps. It uses higher millimeter-wave (MMW) frequencies for the Internet and telecommunications. The IoT system is an essential technology for providing the connection between IoT devices and the human brain. It includes many other paradigms such as artificial intelligence (AI), augmented reality (AR), virtual reality (VR), microchipped and augmented humans, and IoT sensors. Network Slicing is a method of enabling logical, virtualized networks on the same physical network infrastructure in the sustainable 5G environment.

Every network slice is used for particular IoT application services such as healthcare, automotive industry, smart grid, communication, and others, with the help of access and core network functionalities. However, these application services are not small areas, so we categorize them into various function services such as application-based services, platform-based services, and infrastructure-based services. ABS enables the delivery of software applications over the Internet, on-demand, and on a subscription basis. PBS constitutes one of the cloud computing services and on-demand environments for developing, testing, and delivering the application data and the operating system based on the platform. As a cloud-based infrastructure resource, IBS provides essential services such as managing the server, network, data storage, and others. A single network slice is not sufficient for providing these services. To address this problem, we introduce the concept of network sub-slicing in the 5G-enabled IoT with the help of machine learning technology using a novel Machine Learning-based Network Sub-slicing Framework in a sustainable 5G environment.

Figure 2 presents an architectural overview of the proposed network sub-slicing framework in a sustainable 5G environment. It consists of four parts: Physical Resource Network, IoT Platform Layer, Access and Core Network Layer, and IoT Services Layer. The physical resource network is a cloud that

has a data center, and data work is processed in this layer. The physical resource network is divided into various slices such as S1, S2, and S3. In the IoT layer, every slice is categorized into multiple sub-slices using feature selection and clustering, depending on services such as delivering software services, cloud computing services, and cloud-based infrastructure resources services. Feature selection, clustering, or scaling is implemented in the IoT platform layer for processing a particular IoT service. The access and core networks are deployed in the next part as the access and core network layer for providing the functions of the user plane and the control plane. Access network offers various facilities such as centralized storage and communication for local services, storage resources, computing capability and processing, data forwarding, traffic usage reporting, session management, and network slice selection. Thus, load balancing, latency, centralization, computational power, and other issues are resolved in both the IoT services layer and the logical or virtual network layer by using machine learning-based feature selection and clustering or scaling concepts.
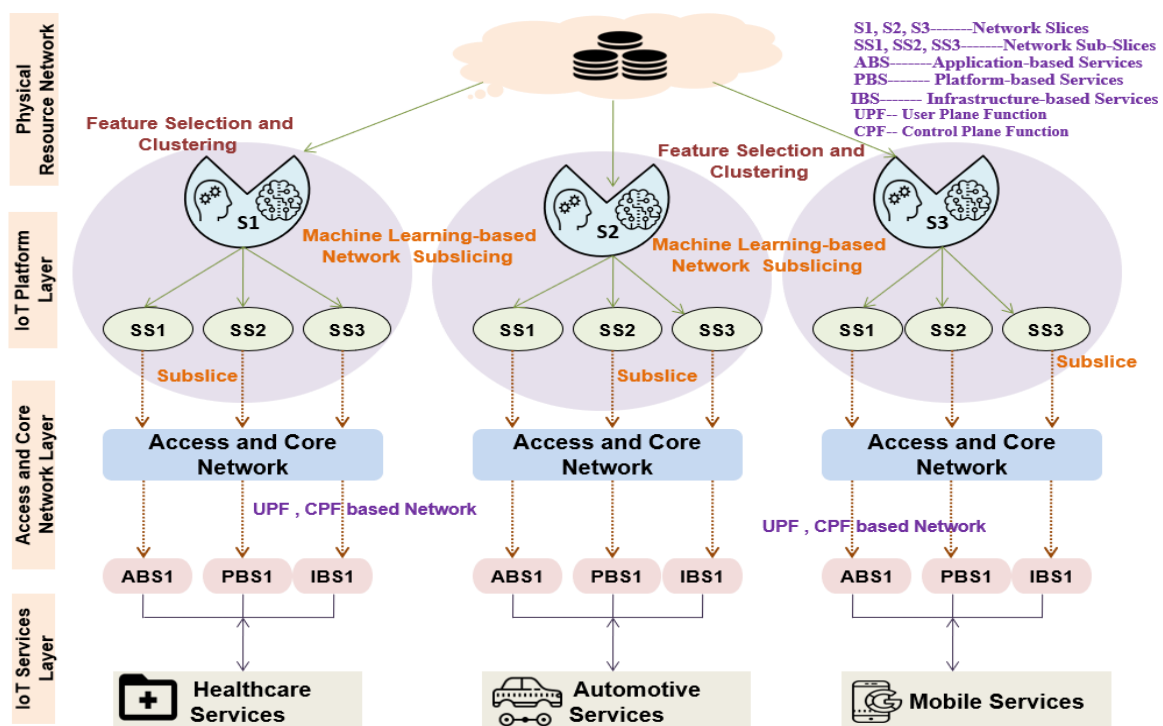

Figure 2. Architectural overview of the proposed network sub-slicing framework in a Sustainable 5G Environment.

The network sub-slicing concept divides the parent network slice to improve the load balancing issue in 5G network slicing. Feature selection and clustering are connected in the network slice at the IoT layer. IoT services are categorized according to the feature selection facility, i.e., application-based, platform-based, and infrastructure-based. These are dependent on software or hardware applications, cloud computing services, and cloud-based infrastructure resources. The SVM algorithm is used for feature selection [27], whereas the K-means algorithm is appropriate for the scaling of processing of IoT services at the IoT layer in the 5G environment [28]. The Access and Core Network is divided into two parts: (1) Access Network and (2) Core Network. These networks provide various facilities in IoT applications. Every sub-slice is applied to particular sub-services in the IoT applications such as application-based, platform-based, and infrastructure-based services in a 5G-enabled sustainable IoT environment.

3.2. Functional Component of the Proposed Network Sub-Slicing Framework

In this subsection, we discuss the main methodological flow of the proposed network sub-slicing framework in the sustainable 5G environment with Figure 3a. It consists of three separate modules and addresses three primary considerations as follows:

(1) Feature selection, which defines how to select the features according to the services of smart applications in the IoT with a sustainable 5G environment through the SVM algorithm.

(2) Feature clustering or scaling, which creates a cluster with the help of the K-means Algorithm and assigns the group based on similar types of application services, such as the application-based, platform-based, and infrastructure-based services (Software applications, cloud computing services, and cloud-based infrastructure resources) of IoT devices.

(3) Access and Core networks, which are used to provide essential functions such as computing of storage, processing, and storage resources, data forwarding, and others.
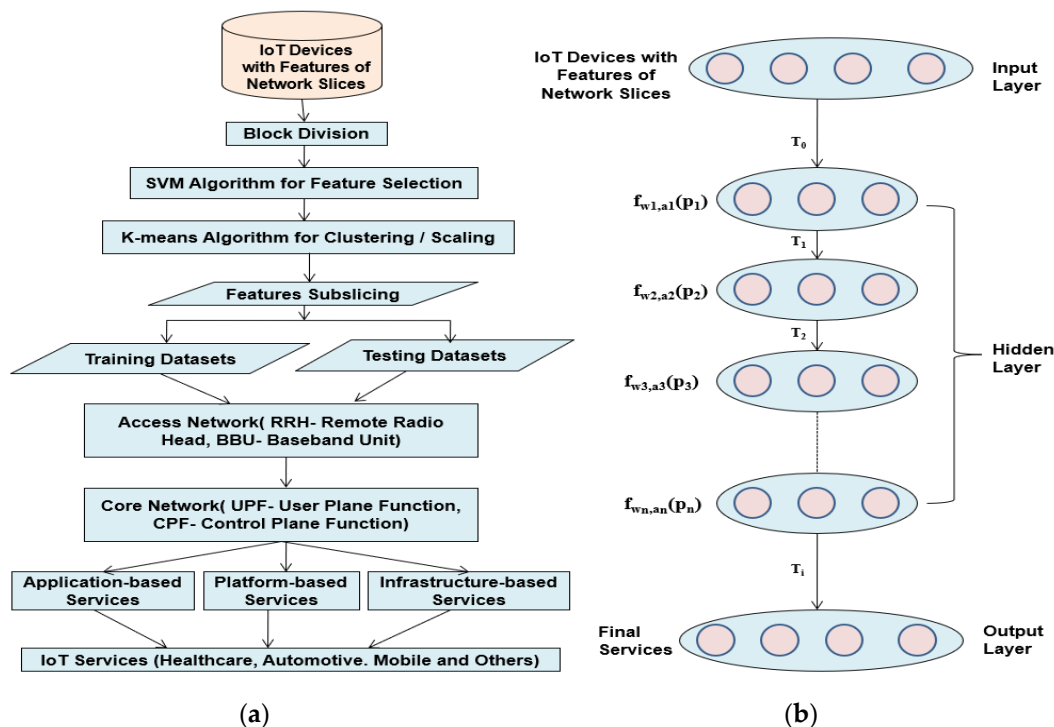

Figure 3. (**a**): Methodological flow of the Machine Learning-based Network Sub-slicing framework. (**b**): Feature Selection and Clustering/Scaling of Devices on the 5G-enabled IoT (Internet of Things) Layer.

The physical network has various input IoT devices with different separated features. Software application features include IP address, packet size, network type, and others. Cloud computing services include developing, testing, and delivering data. Cloud-based infrastructure resources include managing the server, network, and data storage. Thus, these devices are divided into blocks. The SVM algorithm (Algorithm 1) is employed for feature selection based on the smart applications in IoT devices such as smart healthcare, smart vehicle management, and others. The K-means algorithm is utilized for scaling or clustering, and it assigns the group based on similar types of application services. Figure 3b illustrates the internal methodology for feature selection and clustering in sustainable 5G-enabled IoT applications. This methodology comprises three parts, namely the input layer, hidden layer, and output layer. The input layer consists of features such as IP address, packet size, network type, and others for IoT devices. The hidden layer provides the best feature according to the requirements of IoT applications. Application-based, platform-based,

and infrastructure-based services are accessed by the output layer. The modules above are discussed in detail as follows:

Feature selection: Feature selection is the primary function in our proposed framework for network sub-slicing, which is used in the IoT layer. IoT communication devices such as smart mobile, smart computer, cameras, sensors, actuators, and so on are used in IoT layers with network slicing. We are environed by diverse IoT devices and sensors that detect and send data services in many forms—such as audio, video, text, graphics, and others—to the IoT application. IoT data can be measured and monitored in real-time with particular IoT services.

Algorithm 1. Support Vector Machine

Enter the features of training sample IoT devices, such as IP address, network type, packet size, cloud computing services, and cloud-based infrastructure resource features $\{a_i, b_i\}$, with the output-ordered device set defined as B.

Input: The training feature samples set $A = [a_1, a_2, \ldots\ldots a_i, \ldots\ldots a_k]^T$, Class label samples $b = [b_1, b_2, \ldots\ldots b_i, \ldots\ldots b_k]^T$ The feature rank list order set B = Ø.

Output: Projected IoT devices with feature select and weight vector $\{w_k\}$, which is the form of the essential subspace.

Process:

　　1: Subset of surviving features $s = [1, 2, 3, \ldots\ldots\ldots .n]$ and feature rank list B = Ø.
　　2: Feature Sorting repeat until $s = $ Ø
　　3: Restrict training examples to good feature indices: $A_0 = A(:, s)$
　　4: Train the classifier: $\mu = SVM - train(A_0, b)$
　　5: Compute the weight vector to dimension length(s): $w = \sum_k \mu_k b_k a_k$
　　6: Calculate the device ranking with a criteria score value
$$G_k = W_k^2, \qquad \text{Where,} \quad k = 1, 2, 3, \ldots\ldots .|A|$$
$$G_k = \text{Device Ranking with Criteria Score}$$
　　7: Find the features such as bandwidth, weight with the smallest IoT device ranking criteria:
$$F = \arg_k \min G_k$$
　　8: Update the feature set by using the union formula
$$r = [s(f), r] \text{ /* } s = \text{features subset, } r = \text{rank, } f = \text{features}$$
　　9: Destroy this feature with the smallest ranking criteria B such that B = $s(1:f-1, f+1, \text{length}(s))$

Network Slicing is a method that allows the creation of various types of virtual or logical networks inheriting a shared physical infrastructure. It is used in closely related network virtualization technologies that are moving to new network structures such as logical networks toward software-based automation. However, network slicing is not sufficient to provide all types of services to the IoT application. Therefore, network sub-slicing concepts are used by feature selection, providing all types of services, such as application-based, platform-based, and infrastructure-based (software- and hardware-based) services for all particular IoT applications. The SVM algorithm is employed for feature selection in the sustainable 5G-enabled IoT environment. It uses various parameters such as the IP address and port number of the sender and the receiver, size of the network packet, number of packets in the network flow, and network protocol type among others. The SVM algorithm uses a coefficient to rank IoT devices according to the weight or frequency vector created in feature selection by the SVM during the training of devices, and eliminate features attribute to IoT devices with the lowest ranking coefficient in every iteration [29]. Finally, we obtain the decreasing order of all features or service attributes of IoT devices for network sub-slicing. This uses the SVM algorithm to destroy the feature of the service expression device to create a new feature set (collection of IoT devices). The training complexity of the SVM algorithm in the proposed framework is between $O(n^2)$ and $O(n^3)$, with IoT feature services used as training instances because the features are divided according to the status of use of the services in the network.

Feature clustering/scaling: Feature clustering is another part of the proposed framework wherein features are clustered according to their similar properties. A representative feature is selected from each cluster and scaled out to IoT applications such as application-based, platform-based, and infrastructure-based apps and services (i.e., software applications, cloud computing services, and cloud infrastructure resource services). These applications belong to specific forms of IoT, such as smart healthcare, smart transportation, and others. For this function, we use the K-means algorithm.

Algorithm 2. K-means

K is the number of groups (clusters) belonging to the same data points or services, such as software applications, cloud computing, and cloud-based infrastructure resources.

Input: Unordered feature sets are available in IoT devices. Unlabeled data sets are $\{a_1, a_2, a_3, \ldots\ldots a_K\}$, Selected distinct input randomly $\{x_1, x_2, x_3, \ldots\ldots x_M\}$., K is the number of clusters, which has the same data point.

Output: Cluster data centers are $\{a_k\}$ and assignment of the data points are $\{\beta_{mk}\}$, randomly initiate with $\{a_k\}$.

Process:

1:     **Do**
2:         **for** all m = 1 to M **do**
3:         **for** all k = 1 to K **do**   /* m $= 1, 2, 3\ldots\ldots\ldots\ldots\ldots\ldots$ M
4:            **if** $k = \underset{i}{\operatorname{argmin}}(a_i - x_m)^2$ Assign cluster membership
5:                **then** $\beta_{mk} = 1$ /* β_{mk} is set of binary indicator variables
6:             **else** $\beta_{mk} = 0$
7:         **end**
8:         **end**
9:         **for** all k = 1 to K **do**
10:              $a_k = \Sigma_{m=1 \text{ to } M}(x_m * \beta_{mk})/\Sigma_{m=1 \text{ to } M}\beta_{mk}$ / * a_k is the centroid of cluster k, and the average
11:         **end**   of all points assigned to that cluster.
while convergence change the value $\{\beta_{mk}\}$ or $\{a_k\}$
12:     Return $\{a_1, a_2, a_3, \ldots\ldots a_k\}$ and $\{\beta_{mk}\}$
13:     **end procedure**

K-means clustering is one of the simplest and most commonly used unsupervised machine learning algorithms [30]. With the input vector values, the unsupervised algorithm makes inferences in machine learning, but it does not use labels or outcomes for the processing of IoT data. As the main concept of the K-means algorithm, a grouping of similar types of data sets values and identifies underlying patterns. According to the K-means algorithm, we make a cluster of unlabeled data set to provide K number of groups (clusters) belonging to the same data points. These have the same properties, such as software applications, cloud computing services, and cloud-based infrastructure resources. Then, we find the distance between the data points of similar properties. Therefore, finding a set of K cluster centers $\{a_1, a_2, a_3, \ldots\ldots a_k\}$ minimizes the distance between data point values and the nearest data center. To denote the assignment of data point values to the cluster centers, we use a set of binary indicator variables, $\beta_{mk} \in \{0, 1\}$. Therefore, if a data point x_m is assigned to cluster center a_k, then $\beta_{mk} = 1$. Using the equations, we express the problem as follows:

$$Minimize \Sigma_{m=1 \text{ to } M}\Sigma_{k=1 \text{ to } K}\beta_{mk}(x_m - a_k)^2$$

where, $\beta_{mk} = 1$ for data point x_m if it is belongs to cluster K, otherwise $\beta_{mk} = 0$.

$$m = 1, 2, 3, \ldots\ldots\ldots M$$

The minimization has two parts. Firstly, minimize distance with respect to β_{mk} and treat a_k as fixed. Then, minimize distance with respect to a_k and treat β_{mk} as fixed.

The K-means algorithm is significant and highly scalable and is practical for machine learning. It is applied for various functions in IoT fields such as data processing, cluster analysis, and vector quantization. Compared to the Euclidian distance measurement with similar properties, however, this method has certain limitations, such as low accuracy and latency. The algorithm has a loose connection with the k-nearest neighbor classifier. It is a popular machine learning method for classifying IoT data. In the K-means algorithm, we assign a data point value to only one cluster. This may lead to unsuitable clusters in some cases. Applying the 1-nearest neighbor classifier to the cluster centers, we obtain new data values in the existing groups by the *k*-means classifier [31]. The scaling concept used in the proposed framework is shown in Figure 2. Algorithm 2 explains how to receive the optimal cluster data centers $\{a_k\}$ and the authorization of the data points $\{\beta_{mk}\}$. The K-means algorithm has less time complexity in the proposed framework because it assigns the group based on similar types of properties such as software applications, cloud computing services, and cloud-based infrastructure resources. These properties are related to application-based, platform-based, and infrastructure-based services. The time complexity of the K-means algorithm for the proposed framework is $O(n^2)$ because it uses clustering services for nodes.

Access Network and Core Network: The K-means algorithm assigns the group based on similar types of application services such as software applications, cloud computing services, and cloud-based infrastructure resources. Every cluster has some specific frequency, bandwidth, and energy in a combined form. As such, it is scaled in the IoT application according to the specified needs. Therefore, we divide service requirements according to the consumption of energy. Software application services use more power or frequency compared to other services, whereas cloud-based infrastructure resources need less power or frequency. Thus, network sub-slices meet these requirements. IoT devices or communication services such as sensors, Bluetooth, Wi-Fi, and GPRS are connected to the data networks through access and core networks (WLAN, E-UTRAN), providing services to the IoT applications. Access control networks are divided into two parts: (1) Baseband Unit (BBU) and (2) Remote Radio Head (RRH). BBU is employed to provide centralized storage and communications for local sub-slices in access networks. RRH is utilized for resources that are virtualized and isolated to virtual machines by local controllers, providing authentication and authorization to the sub-slice data services of IoT applications. The core networks are further categorized into two parts: (1) User plane function (UPF) and (2) Control plane function (CPF). UPF has various tasks such as traffic reporting and data forwarding in the network in a sustainable 5G environment. CPF also has some services like session management, policy control, mobility management, network slice selection, and authentication server.

After completion of this module, we provided application-based, infrastructure-based, and platform-based services (such as software applications, cloud computing, and cloud-based infrastructure resources) to smart applications (such as smart city, smart transportation, smart farming, and others) by network sub-slicing and utilized the SVM algorithm and the K-means algorithm in the proper methodology of the proposed framework.

4. Experimental Evaluation of the Proposed Network Sub-Slicing Framework

4.1. Evaluation

In this subsection, we present the evaluation of the performance of our proposed Machine Learning-based Network sub-slicing Framework in the sustainable 5G environment in order to provide efficient services to smart applications. We used a powerful workstation with an Intel Xeon Processor E5-1630 v3 (20 MB smart-cache, processor speed of 3.70 Ghz). 5G entails new requirements in terms of bandwidth, latency, network slicing, service maintenance, and so on. For example, the bandwidth requirements have been upgraded from 1GE/10GE to 25GE*N/100 GE. Meanwhile, the latency requirements of the transport node are supported, ranging from 5 μs to 50 μs. The list of tools, equipment, and technologies for the proposed framework and testing are shown in Table 1, together with their respective descriptions. To enable different and separate dedicated logical networks to

customize the respective services, we used the SliceSim simulator. The analysis of various IoT device values used different statistical distributions and observed the effect of usage frequency on bandwidth. The input configurations were utilized by YAML. Python 3.7.7 with Matplotlib, Simpy library, was used to create static, interactive visualizations. The proposed network sub-slicing framework and network slicing framework in the 5G environment were performed and analyzed with the same experiment environment configuration. We evaluated the performance of each method by comparing the transmission delay with several services. We used anonymous MNOs (Mobile Network Operators) operating nationwide data collected offline and stored in the database. The collected data are from the period 2017–2018. Table 2 shows the statistical distribution of the observed test data for all anonymous MNOs operating nationwide data. According to these data, we designed a protocol for the proposed network sub-slicing framework.

Table 1. List of tools, equipment, and technologies for the proposed framework and testing.

S.no.	Equipment Name/Tools/Technology	Description
1.	ICN (Information-Centric Network)	This addresses the Internet's network architectural design gaps based on evolving application requirements.
2.	TelcosRely	This provides a virtual platform like VMware; used for low-level computing resources.
3.	TM500	This provides a potential solution for 5G network slicing test challenges that can validate the network performance for end-users.
4.	Android Mobile Devices	These are the end nodes on which IoT applications run in a sustainable 5G environment.
5.	Microsoft Azure Cloud	Linked to cloud services
6.	Wireless Access Point	A network hub that provides connectivity between IoT devices
7.	OpenStack or Azure Stack	This provides the cloud platform for the 5G environment in IoT applications; used for reset chunks of cloud capacity to commercial clients
8.	MPS (Multimedia Priority Service)	This supports priority sessions on end-to-end network services.
9.	Npm	JavaScript's Node Package Manager
10.	Web3 API	JavaScript API
11.	Extensive Datacenter	For additional sources of revenue besides voice telephony, video value.
12.	3GPP	It has recognized network slicing to be an essential component of 5G.
13.	TeraVM	A software-based tool for 5G application emulation and security.

Table 2. Statistical distribution of observed test data for all MNOs (Mobile Network Operators).

S.no.	Distribution Data According to a Requirement	Value	S.no.	Distribution Data According to a Requirement	Value
1.	No. of MNOs	3	5.	No. of measurements	458,630
2.	No. of districts	942	6.	Average speed of Downlink	42.760 Mbps
3.	No. of cities	80	7.	Average speed of Uplink	17.890 Mbps
4.	Duration	18 months	8.	No. of Items of Test User Equipment	117,050

The Silhouette coefficient is calculated by the mean intra-cluster distance (ε) and the mean nearest-cluster distance (ω) for each sample of IoT device data in the 5G environment with MNOs operating nationwide data. The Silhouette Coefficient for a sample is calculated by $[(\omega - \varepsilon)/\max(\varepsilon, \omega)]$.

For classification, ω is the distance between a sample and the nearest cluster. It is a very important Silhouette coefficient that is defined only if the number of labels is $\left[2 \leq n_{\text{labels}} \leq (n_{\text{samples}} - 1)\right]$. Figure 4a–c show the Silhouette analysis for k-means clustering on MNO 1, 2, and 3, respectively, and plot the graph. We demonstrated the average Silhouette score distribution in MNO 1, 2, and 3; if the K1 = 5 cluster number is selected, then the highest Silhouette coefficient score β = 0.4951 is received. If the K2 = 2 cluster number is selected in MNO-2, then the highest Silhouette coefficient score β = 0.7788 is garnered. If the K3 = 3 cluster number is selected in MNO-3, then the highest Silhouette coefficient score β = 0.5325 is received. Based on this analysis, three network sub-slices show three points concerning the application and services of the IoT in a sustainable 5G environment: (1) High latency with low throughput; (2) High throughput with low latency; (3) Low throughput with low latency. Latency and Heterogeneity are improved by implementing our proposed network sub-slicing framework. In this framework, we studied the bandwidth and latency impact on the parent network slice. We analyzed the Silhouette analysis coefficient (β) using the k value to determine the bandwidth allocation required for each cluster. The value of the cluster group (k) represents the sub-module number of a single application. K1 = 5 means that five sub-modules are required for the application to perform optimally. Based on the Silhouette analysis, the highest β for K1, K2, and K3 is 0.4951, 0.7788, and 0.5325, respectively. Throughput, latency, and bandwidth efficiency are optimized for each K based on the obtained value of β.
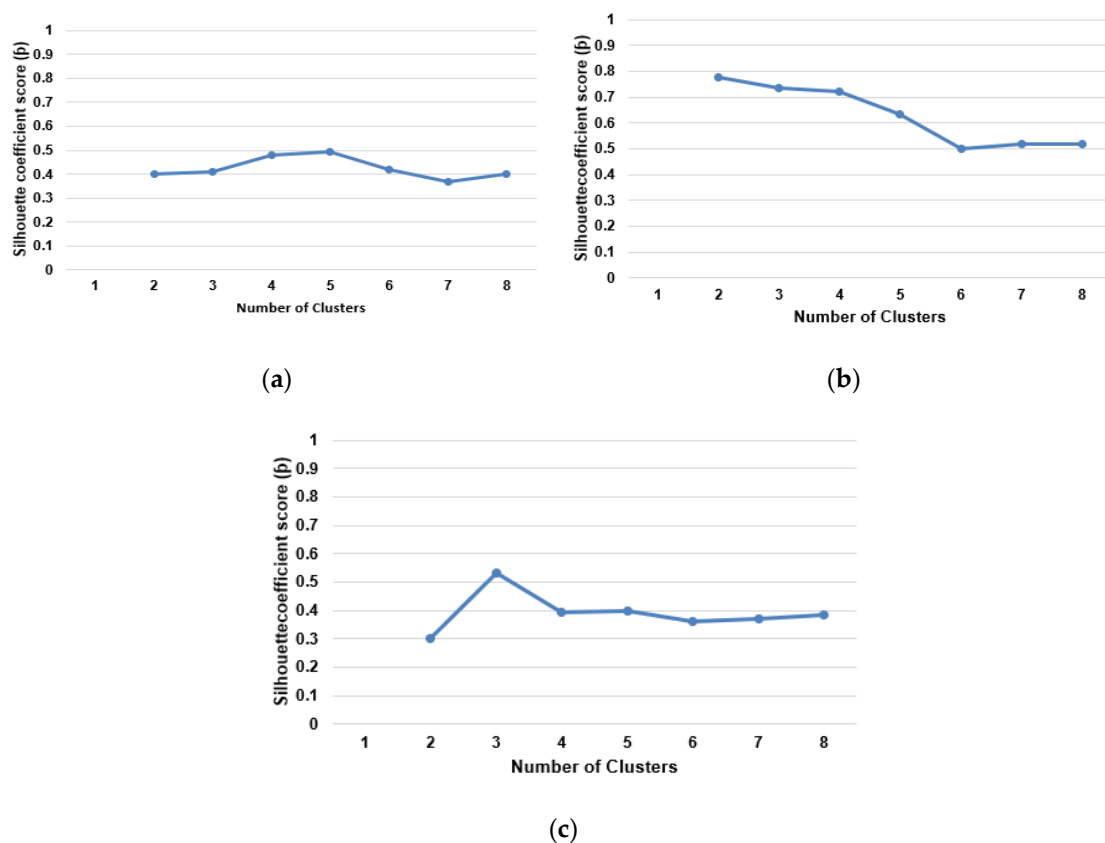

(a)

(b)

(c)

Figure 4. (**a**): Silhouette analysis for k-means clustering on MNO-1. (**b**): Silhouette analysis for k-means clustering on MNO-2. (**c**): Silhouette analysis for k-means clustering on MNO-3.

Figure 5a shows the number of services as the x-axis, and total latency as the y-axis of the proposed framework. A comparison was made to the existing framework or architecture proposed by the RL-NSB Slice Broker of Sciancalepore et al. [16] and the Hierarchical cognitive engine architecture of Hao et al. [18]. We discovered in the comparison that the total transmission delay of both frameworks increased rapidly, but that transmission delay of the proposed framework is shorter than the existing

method. In the network sub-slicing framework, we created the sub-slice from the parent slice and provided the different types of services to smart applications such as healthcare, transportation, and farming in the 5G environment. According to this framework, we could easily reduce the time delay (latency) concerning a number of services in the sustainable 5G environment. The total latency shown in Figure 5a varies from 0 to 700 (ms) with respect to the number of services, 0 to 100, such as software applications, cloud computing services, and cloud-based infrastructure resources, and we compared it with the previous research. Figure 5a is measured based on the count of K obtained. The number of clusters generated is based on the number of sub-modules required for the application's optimum performance.
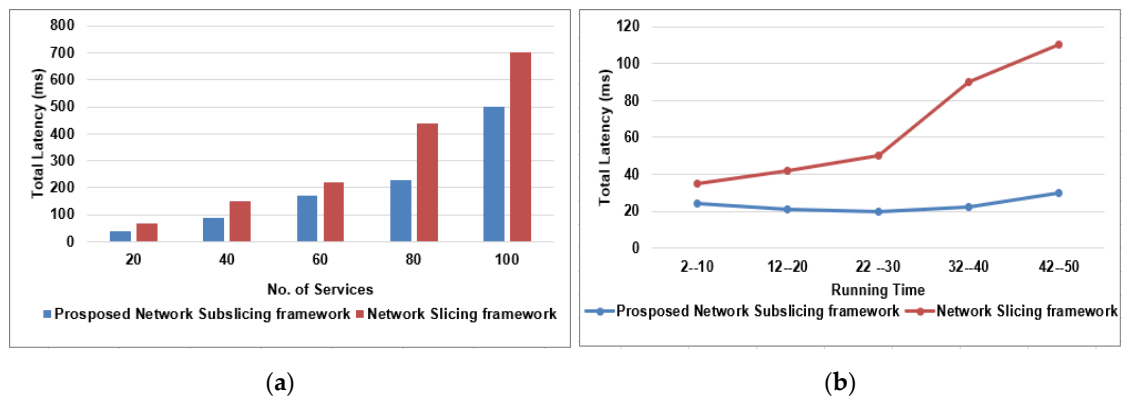

(**a**) (**b**)

Figure 5. (**a**): Comparison of the total latency concerning the number of services. (**b**): Comparison of the total latency concerning the different ranges of services' running time.

Figure 5b shows the comparison of the total latency in msec with respect to the different ranges of services and the running time of the proposed framework and the existing network slicing frameworks according to the different ranges of services' running time of the RL-NSB Slice Broker and the Hierarchical cognitive engine architecture. Running time and total latency use are shown as as x-axis and y-axis, respectively. Overall latency, as shown in Figure 5b, varies from 0 to 120 (ms) with respect to a running time of different ranges of 0 to 50. We compared the total transmission delay using various running time ranges for multiple services such as application-based, infrastructure-based, and platform-based services (software applications, cloud computing, and cloud-based infrastructure resources). This experiment hinged on the premise that the number of services is smaller than the capacity of the networks. The total number of services varies from 0 to 100. Likewise, the capacity or running time of every slice is fixed to 20 ms. In the aspect of running time, the performance of the proposed framework is superior to the existing framework. Total latency in all ranges of the proposed framework's running time is approximately the same, but the existing framework is observed to vary much more in the evaluation. We used a decentralized setting of PoP (point of presence) in virtualized network sub-slices to enable secure communication. Each PoP knew the aggregate workload generated by slice users accessing through sub-slicing. Figure 5b is based on the latency recorded in milliseconds for a single sub-module to operate. Due to the lower bandwidth consumption determined using the silhouette analysis, fewer network parent slice resources are consumed, and the data rate remained high. The time taken in milliseconds for data to travel from the parent slice to the sub-slice is less than that reported in the existing studies.

To determine the spectral and energy efficiency, we identify the minimum energy-efficient sub-slices sets and the relative energy-saving potential. This is calculated using the equation, $(E_{curr} - E_{bench})/E_{curr}$. Where, E_{curr} = Slice's current final energy consumption and E_{bench} = The statistical targeted energy performance of the corresponding slice. We then normalize the current energy consumption, which is related to the Silhouette Coefficient represented by the equation, $(\omega - \varepsilon)/\max(\varepsilon, \omega)$. Figure 6a presents the throughput of the proposed framework for optimal spectrum allocation. As an application

process has dynamic bandwidth requirements, the framework allocates the spectrum based on the varying needs and demands of the process of an application. We found that Equal Allocation (EA) of resources results in bandwidth being wasted. Figure 6a is measured using the varying value of β for a single cluster K. We observe the increase and decrease in β for each cluster, and the throughput of the bandwidth is optimized based on the obtained coefficient value. In contrast, the proposed framework dynamically allocates the frequency, further reducing wastage of the spectrum in the network. Figure 6b shows the spectrum efficiency of the proposed framework compared with the existing network slice structure among multiple processes. We introduced an additional two application processes and compared their performance based on throughput. The saving of bandwidth resources results in higher throughput and contributes to an increase in the signal-to-noise ratio. Figure 6b is measured in bits/second/Hz for different cluster groups, K1, K2, and K3, with the existing network slice technology. The previously observed saving of bandwidth results from using the silhouette coefficient (β), which further contributes to an increase in spectral efficiency or bandwidth efficiency. Thus, the proposed framework consumes fewer bandwidth resources of the network slice, which helps improve the signal-to-noise ratio and throughput. As shown in Figure 6c, the improvement in the rate of throughput has increased energy efficiency using the proposed framework compared with the current 5G network slice energy consumption. The increased performance and reduced energy consumption show that the framework achieves overall power optimization. Figure 6c relies on the optimized throughput obtained from optimally allocating bandwidth on sub-modules. Performance is optimized using the silhouette coefficient value—an increase in throughput results in lower power consumption and greater energy efficiency.
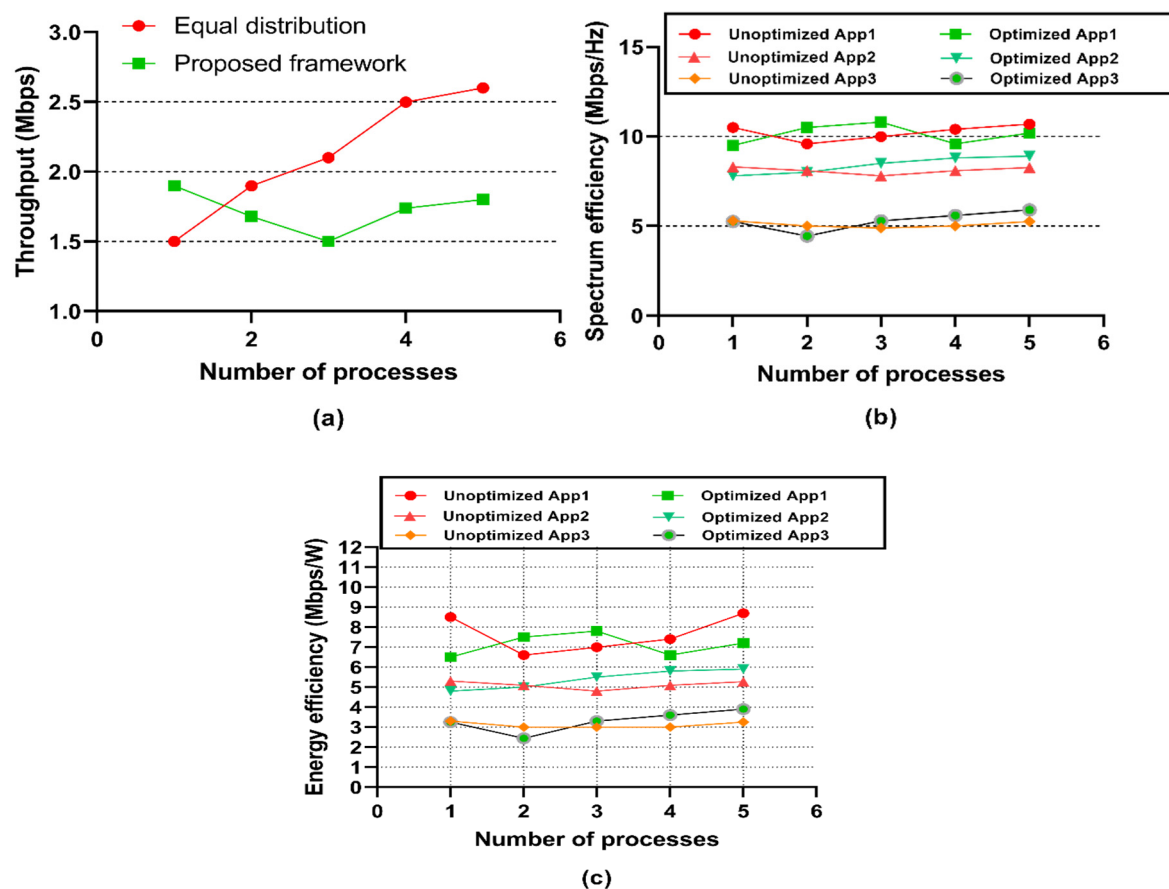

Figure 6. (**a**): Throughput efficiency using a single process, (**b**): Spectrum throughput efficiency using multiple processes, and (**c**): Energy efficiency of the proposed framework.

4.2. Comparative Analysis

The proposed sub-slicing framework is compared with existing research based on three critical factors, as mentioned in Section 2. In the comparative analysis below, which is presented in Table 3, our framework was found to outperform other existing research based on three critical factors: Heterogeneity, Latency, and Load Balancing. We observed that none of the following methods supported the concept of sub-slicing, so they could not provide the flexibility and optimum allocation of resources offered by our proposed sub-slicing framework:

- Heterogeneity: We compared the proposed framework with other methods based on their ability to allocate network slice resources to the application's sub-modules. It compares sub-modules such as software applications, cloud computing services, and cloud-based infrastructure resources with traditional research. Various sub-modules have diverse requirements and different needs from the network. A model capable of addressing the issue of heterogeneous conditions while allocating resources to applications would provide the optimal environment for the system to function.

    1. Proposed framework: Our model first implements a Machine Learning-based Support Vector Machine algorithm to identify the features of different sub-applications. An application such as Healthcare is a broad application with various sub-modules having different needs from other modules. Upon identification of the different features of the modules, we applied the k-means algorithm and cluster common requirements to one group. Groups that require low latency such as medial sensors are grouped in one cluster, and groups requiring high reliability, such as network administrators are arranged as one group. Using the access and core network features of 5G, we further sliced the network and distributed it to each application's sub-module to enhance reliability and efficiency.
    2. RL-NSB Slice Broker: The Reinforcement Learning-based 5G Network Slice Broker implements traffic forecasting using the Holt–Winter (HW) forecasting method. The HW method is used to predict the network slice's traffic based on the historical data of a user's requirement and mobility pattern. Sciancalepore et al. [18] proposed a method that allocates resources based on the needs of different users. Nonetheless, modern applications consist of various sub-modules with heterogeneous sensor devices. The network slice cannot provide optimized services to the whole application functioning on a single allocated network slice.
    3. Hierarchical cognitive engine: Hao et al. [20] proposed a hierarchical cognitive engine (HCE) architecture in order to realize data-driven resource management in 5G networks. The primary aim of their research is to allocate resources to wearable devices based on their requirements. A fitness tracker requires fewer communication resources, whereas AR and VR services have a greater need for communication resources. Their research was focused on providing optimized resources for different heterogeneous devices.
    4. POSENS: Garcia-Aviles et al. [19] proposed an end-to-end network slicing solution for mobile networks. Their research and experimental results were focused on providing isolation for the assigned network slice as well as the ability to offer customized network slices. The POSENS tool offers customizable slices of the network. Still, there is no provision to divide further the slice resource based on different needs. As such, it does not serve the requirements of diverse heterogeneous devices.

- Latency: Latency is a result of the network slice being unable to provide the optimum conditions of data transmission for the devices grouped in the sub-modules [32]. This delay renders the devices unable to connect to their server and provide valuable data in a timely manner. High latency causes the network to malfunction or deliver erroneous feedback. We compared our framework with the existing research literature.

    1. Proposed framework: We are providing division of each slice into sub-slices, which is based on the customized needs of the individual requirements of each sub-module or application. By learning the features of different application sub-modules using SVM and grouping them using k-means, we accurately allocated resources based on their network requirements. Clusters that require low latency in data transmission allocate a sub-slice that fulfills the service level agreement according to those exact requirements.

    2. RL-NSB Slice Broker: The Reinforcement Learning-based slice brokering mechanism allocates resources based on an application's requirements. Every sub-module in the application is assigned the same common network slice configuration as the parent application. Nonetheless, not every sub-module has the same requirements as the parent application, which identifies the broader needs. The RL-NSB proposed model does not optimally address the network needs of each application-based sub-module.

    3. Hierarchical cognitive engine: The proposed HCE architecture implements a data-driven resource management scheme for wearable devices. The cognitive engine analyzes the service features and then uses the cognitive resource engine to monitor the resource consumption of each device. The global cognitive engine allocates resources according to the individual device's needs. Wearables for AR and VR are allocated resources based on their requirements, which helps reduce latency for applications requiring low latency-based solutions.

    4. POSENS: As an end-to-end tool for network slicing based on user/application requirements, POSENS can provide low latency solutions for applications when needed. However, the requirements of each application sub-module are different. While one sub-module requires low latency in data transmission to the server, others may place greater emphasis on spectral efficiency. POSENS does not have a solution based on the separate allocation of resources that serve varying requirements and broad application needs. A standard network slice may serve one sub-module better, but it is not optimized for other sub-modules.

- Load balancing: The deployment of multiple modules of applications has an impact on the performance of the network slice. Each sub-module of an application requires a customizable sub-slice to work in optimized conditions [33–37]. We compared our proposed framework with other methods to assess which exerted the lowest load on the network when operating multiple sub-modules.

    1. Proposed framework: Our model has a separate network sub-slice using machine learning. We divided the finite network resources according to the differing requirements of each sub-module of an application. This step leads to an optimized approach to allocating resources, such as when a sub-module requires low latency. Another sub-module required high reliability to handle the vast deployment of sensor devices. Our proposed model was shown to prevent network slice congestion by dividing network resources further into sub-slices based on their requirements.

    2. RL-NSB Slice Broker: The Reinforcement learning network slice broker broadly addresses the problem of load balancing by allocating resources according to the broad application requirements; nonetheless, it does not address the requirements of each module based on an application's dynamic needs. The slice traffic forecasting feature presented in the RL-NSB method determines the future patterns of network usage and manages load balancing

according to future requirements; but, as an application within itself has varying needs, there is no solution to the problem to optimal load balancing as yet.

3. Hierarchical cognitive engine: The HCE architecture manages load balancing based on the alternating needs of the user's wearable devices. It analyzes the service features and their usage patterns and utilizes service-based resource allocation. The allocation of resources is achieved to enable optimal use of the network resources. Nonetheless, different users have alternating needs, and this method does not address how a single parent slice can address a broad set of users whose network needs differ.

4. POSENS: The proposed POSENS tool is designed to provide customizable network slices based on user requirements. It targets individual network slices for all users or applications and divides the network accordingly. Load balancing is achieved at the broad level of network slicing, and there is no provision to allocate resources optimally for the needs of the individual application's module.

Table 3. Comparative analysis of the proposed framework and existing research.

Methods	Factors			Slicing Mechanism		Key Technology	Need for Data Source
	Heterogeneity	**Latency**	**Load Balancing**	**RAN**	**End-to-End**		
Proposed network sub-slicing framework	Using the SVM and K-means algorithm, sub-slicing is achieved for sub-modules with services.	Clusters that require low latency allocate a sub-slice to fulfill the SLA.	It prevents network slice congestion by further dividing network resources into sub-slices based on their requirements.	Yes	Yes	Machine Learning, IoT, 5G	High
RL-NSB	The proposed method allocates resources for a single network slice. It does not target sub-modules in the application.	The RL-NSB model does not address the low latency needs of sub-modules in applications.	The slice traffic forecasting feature manages load balancing for a single application but not its sub-modules individually.	Yes	No	Reinforcement Learning, 5G	Low
Hierarchical cognitive engine	HCE provides optimized resources for different heterogeneous wearable devices.	The global cognitive engine does allocate resources according to the needs of each individual device.	HCE architecture manages load balancing based on the usage pattern of the user's wearable devices.	No	Yes	Cognitive-based communication	Low
POSENS	It provides customizable slices in the network. There is no provision to further divide the slice resource based on different module requirements.	POSENS does not resolve the separate allocation of resources that serve varying requirements and broad application needs.	Load balancing is achieved at the broad level of network slicing, but there is no provision to allocate resources optimally to meet the needs of the sub-modules.	No	Yes	Radio Access Networks	High

5. Conclusions

In this paper, we have proposed a machine learning-based network sub-slicing framework in the sustainable 5G environment and provided sustainable solutions for network sub-slicing. According to the proposed framework, sub-slicing concepts are presented for optimizing network and load balancing, including power efficiency problems wherein each logical slice is divided into a virtualized sub-cluster of resources. Each sub-slice provides the application system with different prioritized required resources. The SVM algorithm and the K-Means algorithm are employed in the proposed framework for feature selection and clustering, respectively, to enable efficient communication to the IoT application in the sustainable 5G networks. We present an analysis of the proposed network sub-slicing framework, which is shown to outperform existing studies with three considerations: Heterogeneity, Latency and Load Balancing, and Power Efficiency. We compared the power and throughput efficiency using single and multiple processes with the existing 5G network slicing technology. The framework demonstrated improved performance and reduced energy consumption. It shows the overall achievement of power optimization and efficient performance with power consumption. An experimental analysis was performed, demonstrating the effectiveness of the proposed framework compared to existing research.

In this paper, we have provided a conceptual design of a system prototype and a concise evaluation based on the strengths of the individual solutions adopted. Future endeavors will be focused on using the developed network sub-slicing to evaluate the successful operation of the proposed framework and to optimize the different parameters involved and controlled in the large node in the 5G environment. The vulnerability of the proposed framework is related to the increasing numbers and types of cyberattacks, which could affect the performance of a sub-slice, and consequently, the parent network slice. Future research will address the security implications arising from the implementation of the proposed sub-slicing framework.

Author Contributions: Conceptualization, S.K.S., and M.M.S.; Methodology, S.K.S.; Validation, S.K.S., M.M.S.; Formal analysis, M.M.S. and J.C.; Investigation, S.K.S.; Resources, M.M.S. and J.C., Data curation, M.M.S.; Writing—review & editing, S.K.S., and M.M.S.; Writing—original draft, S.K.S. and M.M.S.; Visualization, M.M.S., Supervision, J.H.P.; Project administration, J.H.P.; Funding acquisition, Y.P. and J.H.P. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Arfaoui, G.; Bisson, P.; Blom, R.; Borgaonkar, R.; Englund, H.; Félix, E.; Papay, J. A security architecture for 5G networks. *IEEE Access* **2018**, *6*, 22466–22479. [CrossRef]
2. How 5G Help Fuel the Next Generation of IoT Projects. Available online: https://www.iotworldtoday.com/2019/03/18/how-5g-could-help-fuel-the-next-generation-of-iot-projects/ (accessed on 20 March 2019).
3. Alqarni, M.A.; Chauhdary, S.H.; Malik, M.N.; Ehatisham-ul-Haq, M.; Azam, M.A. Identifying smartphone users based on how they interact with their phones. *Hum. Cent. Comput. Inf. Sci.* **2020**, *10*, 1–14. [CrossRef]
4. Afolabi, I.; Taleb, T.; Samdanis, K.; Ksentini, A.; Flinck, H. Network slicing and softwarization: A survey on principles, enabling technologies, and solutions. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2429–2453. [CrossRef]
5. Zhang, H.; Liu, N.; Chu, X.; Long, K.; Aghvami, A.H.; Leung, V.C. Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges. *IEEE Commun. Mag.* **2017**, *55*, 138–145. [CrossRef]
6. Danial, S.N.; Smith, J.; Veitch, B.; Khan, F. On the realization of the recognition-primed decision model for artificial agents. *Hum. Cent. Comput. Inf. Sci.* **2019**, *9*, 36. [CrossRef]
7. Kotulski, Z.; Nowak, T.W.; Sepczuk, M.; Tunia, M.; Artych, R.; Bocianiak, K.; Osko, T.; Wary, J.P. Towards constructive approach to end-to-end slice isolation in 5G networks. *Eurasip J. Inf. Secur.* **2018**, *1*, 2. [CrossRef]
8. Li, S.; Da Xu, L.; Zhao, S. 5G Internet of Things: A survey. *J. Ind. Inf. Integr.* **2018**, *10*, 1–9. [CrossRef]

9.　Mamolar, A.S.; Pervez, Z.; Calero, J.M.A.; Khattak, A.M. Towards the transversal detection of DDoS network attacks in 5G multi-tenant overlay networks. *Comput. Secur.* **2018**, *79*, 132–147. [CrossRef]

10.　Chekired, D.A.; Togou, M.A.; Khoukhi, L.; Ksentini, A. 5G-Slicing-Enabled Scalable SDN Core Network: Toward an Ultra-Low Latency of Autonomous Driving Service. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 1769–1782. [CrossRef]

11.　Shurman, M.; Taqieddin, E.; Oudat, O.; Al-Qurran, R.; Al Nounou, A.A. Performance Enhancement in 5G Cellular Networks Using Priorities in Network Slicing. In Proceedings of the 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), IEEE, Amman, Jordan, 9–11 April 2019; pp. 822–826. [CrossRef]

12.　Gupta, R.; Tanwar, S.; Tyagi, S.; Kumar, N. Machine learning models for secure data analytics: A taxonomy and threat model. *Comput. Commun.* **2020**, *153*, 406–440. [CrossRef]

13.　Perez, A.J.; Zeadally, S.; Jabeur, N. Security and Privacy in Ubiquitous Sensor Networks. *J. Inf. Process. Syst.* **2018**, *14*, 286–308. [CrossRef]

14.　Tkachenko, R.; Izonin, I.; Kryvinska, N.; Dronyuk, I.; Zub, K. An Approach towards Increasing Prediction Accuracy for the Recovery of Missing IoT Data based on the GRNN-SGTM Ensemble. *Sensors* **2020**, *20*, 2625. [CrossRef] [PubMed]

15.　Singh, S.K.; Rathore, S.; Park, J.H. BlockIoTIntelligence: A Blockchain-enabled Intelligent IoT Architecture with Artificial Intelligence. *Future Gener. Comput. Syst.* **2019**, *110*, 721–743. [CrossRef]

16.　Beshley, M.; Kryvinska, N.; Seliuchenko, M.; Beshley, H.; Shakshuki, E.M.; Yasar, A.-U.-H. End-to-End QoS "Smart Queue" Management Algorithms and Traffic Prioritization Mechanisms for Narrow-Band Internet of Things Services in 4G/5G Networks. *Sensors* **2020**, *20*, 2324. [CrossRef] [PubMed]

17.　Nour, B.; Ksentini, A.; Herbaut, N.; Frangoudis, P.A.; Moungla, H. A Blockchain-Based Network Slice Broker for 5G Services. *IEEE Netw. Lett.* **2019**, *1*, 99–102. [CrossRef]

18.　Sciancalepore, V.; Costa-Perez, X.; Banchs, A. RL-NSB: Reinforcement Learning-Based 5G Network Slice Broker. *IEEE ACM Trans. Netw.* **2019**, *27*, 1543–1557. [CrossRef]

19.　Garcia-Aviles, G.; Gramaglia, M.; Serrano, P.; Banchs, A. POSENS: A practical open source solution for end-to-end network slicing. *IEEE Wirel. Commun.* **2018**, *25*, 30–37. [CrossRef]

20.　Hao, Y.; Jiang, Y.; Hossain, M.S.; Ghoneim, A.; Yang, J.; Humar, I. Data-driven Resource Management in a 5G Wearable Network using Network Slicing Technology. *IEEE Sens. J.* **2018**, *19*, 8379–8386. [CrossRef]

21.　Dighriri, M.; Alfoudi, A.S.D.; Lee, G.M.; Baker, T.; Pereira, R. Resource allocation scheme in 5G network slices. In Proceedings of the 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), Krakow, Poland, 16–18 May 2018; pp. 275–280. [CrossRef]

22.　Kasgari, A.T.Z.; Saad, W. Stochastic optimization and control framework for 5G network slicing with effective isolation. In Proceedings of the 52nd Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 21–23 March 2018; pp. 1–6. [CrossRef]

23.　Bega, D.; Gramaglia, M.; Fiore, M.; Banchs, A.; Costa-Perez, X. DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 280–288. [CrossRef]

24.　Abbas, M.T.; Khan, T.A.; Mahmood, A.; Rivera, J.J.D.; Song, W.C. Introducing network slice management inside m-cord-based-5g framework. In Proceedings of the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018; pp. 1–2. [CrossRef]

25.　Kafle, V.P.; Fukushima, Y.; Martinez-Julia, P.; Miyazawa, T. Consideration on automation of 5G network slicing with machine learning. In Proceedings of the 2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K), Santa Fe, Argentina, 26–28 November 2018; pp. 1–8. [CrossRef]

26.　Le, L.V.; Lin, B.S.P.; Tung, L.P.; Sinh, D. SDN/NFV, Machine Learning, and Big Data Driven Network Slicing for 5G. In Proceedings of the 2018 IEEE 5G World Forum (5GWF), Silicon Valley, CA, USA, 9–11 July 2018; pp. 20–25. [CrossRef]

27.　Kuang, F.; Zhang, S.; Jin, Z.; Xu, W. A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection. *Soft Comput.* **2015**, *19*, 1187–1199. [CrossRef]

28.　Tang, Z.; Liu, K.; Xiao, J.; Yang, L.; Xiao, Z. A parallel k-means clustering algorithm based on redundance elimination and extreme points optimization employing MapReduce. *Concurr. Comput. Pract. Exp.* **2017**, *29*, e4109. [CrossRef]

29. Kuang, F.; Xu, W.; Zhang, S. A novel hybrid KPCA and SVM with GA model for intrusion detection. *Appl. Soft Comput.* **2014**, *18*, 178–184. [CrossRef]

30. Singh, S.K.; Jeong, Y.S.; Park, J.H. A Deep Learning-based IoT-oriented Infrastructure for Secure Smart City. *Sustain. Cities Soc.* **2020**, 102252. [CrossRef]

31. Lee, D.; Park, J.H. Future Trends of AI-Based Smart Systems and Services: Challenges, Opportunities, and Solutions. *J. Inf. Process. Syst.* **2019**, *15*, 717–723. [CrossRef]

32. Rathore, S.; Park, J.H. Semi-supervised learning based distributed attack detection framework for IoT. *Appl. Soft Comput.* **2018**, *72*, 79–89. [CrossRef]

33. Singh, S.K.; Salim, M.M.; Cho, M.; Cha, J.; Pan, Y.; Park, J.H. Smart Contract-Based Pool Hopping Attack Prevention for Blockchain Networks. *Symmetry* **2019**, *11*, 941. [CrossRef]

34. Chauhan, K.; Jani, S.; Thakkar, D.; Dave, R.; Bhatia, J.; Tanwar, S.; Obaidat, M.S. Automated Machine Learning: The New Wave of Machine Learning. In Proceedings of the 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), IEEE, Bangalore, India, 5–7 March 2020; pp. 205–212. [CrossRef]

35. Yin, C.; Ding, S.; Wang, J. Mobile marketing recommendation method based on user location feedback. *Hum. Cent. Comput. Inf. Sci.* **2019**, *9*, 14. [CrossRef]

36. Liu, M.; Cheng, L.; Qian, K.; Wang, J.; Wang, J.; Liu, Y. Indoor acoustic localization: A survey. *Hum. Cent. Comput. Inf. Sci.* **2020**, *10*, 2. [CrossRef]

37. Wang, J.; Gu, X.; Liu, W.; Sangaiah, A.K.; Kim, H.J. An empower hamilton loop based data collection algorithm with mobile agent for WSNs. *Hum. Cent. Comput. Inf. Sci.* **2019**, *9*, 1–4. [CrossRef]